

2

SECURITY CLASSIFICATION OF THIS PAGE

PORT DOCUMENTATION PAGE

AD-A203 241

1D. RESTRICTIVE MARKINGS

OTIC FILE COPY

3. DISTRIBUTION/AVAILABILITY OF REPORT

2D. DECLASSIFICATION/DOWNGRADING SCHEDULE

UNLIMITED

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

5. MONITORING ORGANIZATION REPORT NUMBER(S)

AFOSR-TR-88-1288

6a. NAME OF PERFORMING ORGANIZATION

6b. OFFICE SYMBOL
(If applicable)

7a. NAME OF MONITORING ORGANIZATION

Univ of California

AFOSR/NE

6c. ADDRESS (City, State and ZIP Code)

LaJolla, CA 92093

7b. ADDRESS (City, State and ZIP Code)

Bldg 410
Bolling AFB DC 20332-6448

8a. NAME OF FUNDING/SPONSORING
ORGANIZATION

8b. OFFICE SYMBOL
(If applicable)

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

AFOSR/NE

AFOSR-88-0022

8c. ADDRESS (City, State and ZIP Code)

Bldg 410
Bolling AFB DC 20332-6448

10. SOURCE OF FUNDING NOS.

PROGRAM
ELEMENT NO.

PROJECT
NO.

TASK
NO.

WORK UNIT
NO.

61102F

DARPA

00

11. TITLE (Include Security Classification)
Architecture Studies and System Demonstrations
Parallel Processor for AI and NI

of Optical

6150

12. PERSONAL AUTHOR(S)

Dr. Lee

13a. TYPE OF REPORT

SemiAnnual

13b. TIME COVERED

FROM 1 Oct 88 TO

14. DATE OF REPORT (Yr., Mo., Day)

15. PAGE COUNT

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD GROUP SUB. GR.

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

During the last six months we have applied the results of our studies on existing parallel computing architectures for AI and NI to develop the Programmable Opto-Electronic Multiprocessor (POEM) architecture. Our goal was design a scalable architecture suitable for AI and ultimately for NI that will take full advantage of the hybrid nature of opto-electronic technologies. In the POEM system this is achieved by implementing all communication using photonics and all logic their local interconnections using electronics.

DTIC
ELECTE
DEC 16 1988
a
E

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT

UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS ☐

21. ABSTRACT SECURITY CLASSIFICATION

UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL

GILES

22b. TELEPHONE NUMBER
(Include Area Code)

(202) 767-4931

22c. OFFICE SYMBOL

NE

AFOSR-TR. 88-1288

Semiannual Technical Report*

for

Architecture Studies and System Demonstrations of Optical Parallel

Processor for AI and NI

October 1, 1988

Approved for public release;
distribution unlimited.

Sponsored by

Defense Advanced Research Projects Agency

DARPA Order No. 6150, Code 7D10

Monitored by AFOSR Under Grant No. AFOSR-88-0022

Grantee

The Regents of the University of California

University of California, San Diego

La Jolla, CA 92093

Principal Investigators:

Sing H. Lee

Sadik C. Esener

(619) 534-2413

(619) 534-2732

Program Manager:

Dr. C. L. Giles

(202) 767-4931

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

REPORT OF SCIENTIFIC RESEARCH (AFSC)
 TITLE: RESEARCH REPORT NO. 10 TO DTIC
 AUTHOR: RESEARCH REPORT NO. 10 TO DTIC
 SUBJECT: RESEARCH REPORT NO. 10 TO DTIC
 D. RESEARCH REPORT NO. 10 TO DTIC
 MAINTAINED BY RESEARCH REPORT NO. 10 TO DTIC
 Chief, Technical Information Division

SUMMARY

During the last six months we have applied the results of our studies on existing parallel computing architectures for AI and NI to develop the Programmable Opto-Electronic Multiprocessor (POEM) architecture. Our goal was to design a scalable architecture suitable for AI and ultimately for NI that will take full advantage of the hybrid nature of opto-electronic technologies. In the POEM system this is achieved by implementing all communication using photonics and all logic and their local interconnections using electronics.

The POEM system has a highly parallel architecture based on wafer-scale integration of opto-electronic processing elements and programmable free-space optical interconnects. Although the POEM architecture can support any grain size, synchrony and topology, our first design has focussed on a fine-grain SIMD POEM machine containing a very large number of simple 1-bit silicon PEs in order to match the requirements of efficient parallel algorithms in AI. Because the optical interconnects of POEM are programmable the communication overhead among PE's is greatly reduced when compared to existing electronic fine-grain machines such as the Connection Machine. Also optical interconnects enable the size and the weight of POEM systems to be considerably less.

In order to extract the true value of POEM systems we have compared their computational ability and technological feasibility to that of an all optical symbolic substitution (SS) system. The results of this comparison indicated that POEM systems are more feasible and computationally more efficient than symbolic substitution systems; the results have been reported at the 1988 Annual Meeting of the Optical Society of America at Santa Clara and will be published in the special issue of Optical Engineering, March 1989.

1. Introduction

During the last six months we continued our study with Professor M. Paturi of the UCSD Computer Sciences Department on various opto-electronic architectures for AI. These studies led to the development of a new opto-electronic computing architecture that we call the programmable

opto-electronic multiprocessor system (POEM). We have designed a fine grain POEM system and compared its computational abilities to that of symbolic substitution systems. In the next sections we will briefly describe the POEM architecture that was designed for solving AI problems and report on our comparison results.

2. The POEM system

The POEM system has a highly parallel architecture based on wafer scale integration of opto-electronic processing elements and programmable free-space optical interconnects. The POEM system will be realized with an integrated silicon/electro-optic modulator technology (e.g., Si/PLZT) to implement the PE arrays, and 3-D holographic storage medium such as photorefractive crystals as reported in the last semi-annual report to support the programmable interconnects. This storage capability makes it possible to retrieve prestored interconnections patterns connecting the processing elements at speeds compatible with the system clock rate (1MHz). Therefore, programmable optical interconnects enable communication in POEM system without the need for message passing while minimizing the complexity of the electronic processing elements. The architecture of POEM can support any variation of the parameters commonly used to classify parallel architectures: granularity (fine, coarse or large grain), synchrony (SIMD or MIMD) and topology.

We have determined that efficient parallel algorithms in symbolic computing require a very large number of highly interconnected but simple PE's. Therefore, our design has focussed on a fine-grain SIMD POEMs machine containing a very large number (100,000 or more) of simple 1-bit silicon PEs. An opto-electronic controller is used to optically broadcast an instruction stream to the PEs for SIMD processing. Unlike conventional parallel systems, there is no fixed interconnection topology among the processors. Instead, the programmable optical interconnects are determined by the opto-electronic controller. The programmer implements a topology that best matches his algorithm.

Experimentally, we have designed a scalable prototype of such POEM machines with 8 PEs. Each PE has a 64 bit RAM, several registers, including an accumulator, a carry register and a sleep

register, and can perform logic (AND, OR, complement and ADD operations), data movement, conditional execution, memory and I/O operations. We plan to fabricate these PEs using a hybrid silicon/PLZT technology. The instruction set and the timing have also been determined, and will be broadcast optically to the control unit associated with the PEs. Among a wide variety of algorithms to which it was designed to apply parallelism the fine-grain POEMs machine is especially effective for the rapid execution of symbolic information processing tasks and graph algorithms because of the programmability of optical interconnects and the large number of simple PEs. In particular, it is expected to offer extremely high performance in the rapid execution of semantic networks, production systems, management of large knowledge bases, parallel databases, transportation and communication optimization problems and computer-aided-design.

3. Comparison of POEM system with symbolic substitution machines

Our purpose in performing a quantitative comparison between symbolic substitution and POEM was to extract the true value of POEM systems. The comparison is based on the computational ability and technological feasibility of both systems.

First, we uncovered that the computational ability of an optical SS system is essentially equivalent to a 2-D VLSI SIMD mesh connected array of small-grain processors. We can show that a SS rule can be simulated by a mesh of electronic processors, using only a small number of cycles depending on the complexity of the rule. In addition, the simulation of even a very small-grain processor mesh in SS seems to require more space and time. Furthermore, SS lacks any means of implementing a RAM function because of its local interconnection topology. This implies that space-time tradeoffs are hard to achieve on SS machines. On the other hand, as mentioned in the previous section the POEM system can implement any variation of the three parameters used to classify parallel systems. By mapping commonly used algorithms such as FFT, sorting, graph optimization, etc. onto a SS and POEMs architectures with global interconnection topology (e.g., hypercube), we have determined that the SS approach (operated at very high clock rates, e.g., above 500 MHz) is outperformed by the POEMs (operated at much slower clock rates, e.g., 10 MHz). Therefore, the speed of the devices used in SS systems is offset by the inefficiency

of the algorithms used on a locally interconnected topology.

We have also considered the technological limitations involved in constructing the SS and the POEMs machines. We can provide quantitative measures for such important technological characteristics as speed, power dissipation and area for SS and POEMs architectures. For SS we have directly related these characteristics to the complexity of the SS rule, and taken note of the fact that more complex rules will require larger dynamic range from the thresholding devices in the recognition stage. We also compared the energy loss involved in implementing a simple boolean function using POEMs and SS paradigm, recognizing that the major energy loss in SS paradigm is introduced by the thresholding devices. Based on the above analysis, we conclude that in the foreseeable future the POEM system is better suited for general purpose digital optical computing as compared to SS, due to its computational ability and technological feasibility. More detail about this comparison can be found in the attached reference that was submitted to Optical Engineering.

4. Conclusions and future directions

During the last six months we have developed a new architecture, the POEM system that takes full advantage of its hybrid technology and programmable optical interconnects in order to solve AI problems efficiently. We have evaluated its computational abilities and technological feasibility by comparing POEM to symbolic substitution systems.

We are currently comparing POEM to existing electronic fine grain machines and mapping well known knowledge-base systems such as the NETL onto POEM system. We are also presently implementing experimentally the phase coded matrix tensor multiplier to demonstrate the programmability of optical interconnects.

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	



A Comparison of Programmable Opto-Electronic Multiprocessors and Symbolic Substitution for Digital Optical Computing

F. Kiamilev, S. Esener, R. Paturi, Y. Fainman, P. Mercier, C. Guest, and S.H. Lee
Electrical and Computer Engineering Department
University of California, San Diego
La Jolla, CA 92093

ABSTRACT

This paper compares symbolic substitution systems with arrays of optically interconnected electronic processors. The comparison is made on the bases of computational efficiency, speed, size, energy utilization, programmability, and fault-tolerance. The small grain size and space invariant connections of symbolic substitution lead to poor computational efficiency, difficult programming, and difficult incorporation of fault tolerance. Reliance on optical gates as its fundamental building elements is shown to give poor energy utilization. Programmable Opto-Electronic Multiprocessors (POEMs), on the other hand, provide the architectural flexibility for good computational efficiency, use an energy efficient combination of technologies, and support traditional programming methodologies and fault tolerance. Though the inherent clock speed of POEMs is slower than that of symbolic substitution systems, for most problems they will provide greater computational throughput.

1. Introduction

The planar nature of electronic VLSI technology imposes limits on parallel electronic computing interconnect latency and area.¹ Free-space optically interconnected processing elements offer an opportunity to remove this limitation by providing interconnections in three dimensions.²⁻⁴ We describe here general-purpose computing systems currently under investigation at UCSD that integrate opto-electronic processing elements and free-space programmable optical interconnects. These systems combine the advantages of efficient processing abilities of silicon technology and programmable global communication provided by optical interconnects. We call these systems Programmable Opto-Electronic Multiprocessors (POEMs).⁵

To place the characteristics of POEMs in context, we will compare them to an alternative general purpose optical computing system based on symbolic substitution (SS) that has been presented by Huang et. al.^[6-8] Both POEMs and SS are being proposed for achieving high performance, general purpose, and parallel computing. In this paper we examine the performance potentials and technological limits of these two systems. The evaluation of these systems will be based on their ability to implement various algorithms efficiently, the power and area requirements of existing and projected technologies to implement them, fault tolerance, and ease of programming.

Section 2 provides architectural descriptions as well as example implementations of POEMs and symbolic substitution systems. In section 3, we establish the computational equivalence of SS systems to a 2-dimensional mesh of VLSI processors. Technological considerations are discussed in section 4, including system size, speed and energy dissipation. In section 5 the relative merits of SS and POEMs systems are compared. Section 6 presents our conclusions.

2. Summary Descriptions of POEMs and SS

In this section we describe briefly the architectures and fundamental features of POEMs and SS. Specific characteristics important for the comparison of the systems are emphasized.

2.1. POEMs Architecture

2.1.1. Architecture Description

Programmable Opto-Electronic Multiprocessors have a highly parallel architecture based on wafer scale integration of opto-electronic processing elements (PEs) and reconfigurable free-space optical interconnects. The POEMs machine can be realized with an integrated opto-electronic technology, such as silicon/PLZT^{9,10} for the PE arrays, and dichromated gelatin as the volume holographic storage medium for the interconnects. The POEMs architecture can be extended to be reprogrammable or reconfigurable using a real-time volume holographic medium such as photorefractive crystals.

The POEMs architecture uses electrical interconnects for local communication within a PE and holographic optical interconnects for global communication among PEs. As shown in reference 11 for interconnections longer than a certain break-even length free space holographic optical interconnects consume less energy and are faster than their electrical counterparts. Also, free space interconnects are immune to the crossover constraints of planar electronic technology, allowing denser interconnection topologies. Furthermore, they release space in the processing planes used for interconnects, allowing more silicon circuitry on the wafer. The POEMs machines use light modulators as optical transmitters. When compared to active light sources, such as lasers or LED's, light modulators are attractive because i) they may be easier to integrate with silicon and ii) they dissipate less power on-wafer, since electrical to optical conversion power is dissipated off-wafer. This also allows on-wafer power dissipation to be independent of the fanout of the processor communication network, if electro-optic light modulators are used.

The POEMs architecture can support any variation of the parameters commonly used to classify parallel architectures: granularity (fine, coarse or large grain), synchrony (SIMD or MIMD) and topology. The strength of POEMs machines comes from their efficient implementation of interconnections and the large degree of parallelism and connectivity that is inherent in free-space programmable global optical interconnections.

2.1.2. Implementation

As an example, we now describe a fine-grain POEMs machine (Fig.1.a.) containing a very large number (100,000 or more) of simple 1-bit silicon processors. An opto-electronic controller, connected to a sequential host computer, is used to optically broadcast the instruction stream and the master clock through a computer generated hologram to the PEs for SIMD processing. The global inter-processor communication in POEMs is implemented by activating different interconnection holograms in a volume holographic material of large storage capacity such as dichromated gelatin. Each interconnection hologram is recorded with a different random phase code. These holograms can be activated independently at speeds compatible with the system clock rate by displaying the appropriate random phase code on a small SLM. Therefore, unlike conventional parallel systems, there are no limitations from fixed interconnection topology among the processors. Instead, the programmable optical interconnects are determined by the opto-electronic controller. Therefore, the programmer can implement a topology that best matches the current algorithm. In addition, the interconnection storage capacity requirement on the holographic material can be reduced if real-time

reprogrammable material requirements can be added. For example, one may envision using photorefractive crystals or other non-linear optical materials to apply reprogrammable interconnects to the PEs. In this case, the user will be capable of reconfiguring the POEMs in a very short time to match his algorithmic requirements.

The internal data paths of the PE's are implemented electrically as in a common electronic processor. Each PE has the capability to perform logic, conditional execution, data movement, and I/O operations (Fig.1b.). Also, each PE has some local RAM to support the conventional programming models. In general, the grain size of the PEs is governed by the break-even interconnection distance found by equating the energy required by the local and global interconnects, and by the computational and concurrency requirements imposed by a given application. For some applications, the amount of required memory governs the grain size of the PE, resulting in non-scalable systems. In POEMs, the physical size of a PE may be governed by the size of the RAM even for a small number of storage cells. However, a RAM function is crucial for performing context switching, that is for handling a number of processes larger than the number of PEs in the system. Optical memory systems that will support large memory bandwidth and large storage capacity will remove these limitations and increase the range of application of POEMs.

The fine-grain POEM machine was designed to apply parallelism to a wide variety of algorithms. However, because of the programmability of optical interconnects and the large number of simple processing elements, it is particularly effective for the rapid execution of symbolic information processing tasks and graph algorithms. The fine-grain POEM machine is expected to offer flexibility and high performance in the rapid execution of semantic networks, production systems, management of large knowledge bases, transportation and communication optimization problems, computer aided-design, VLSI circuit simulation, parallel databases and game playing.

2.2. Symbolic Substitution Based Computing Systems

In this section, we give a brief review of symbolic substitution based computing systems and some of the proposed optical implementations.

2.2.1. Architecture Description

The idea of symbolic substitution is derived from cellular automata considered by Von Neumann¹² in which locally interconnected cells evolve using certain transition rules. The motivation for considering such computational models is the desire to show that a collection of locally interconnected devices (cells) governed by simple transition rules can exhibit interesting computational properties.

Symbolic substitution is an elaboration on the idea of cellular automata, suited for optical implementation.⁷ Symbolic substitution is a pattern rewriting procedure that operates in a parallel and space-invariant fashion on a two-dimensional plane of binary pixels. Every occurrence of a given pattern is replaced by another pattern. Each such pair of patterns is called a substitution or a transition rule. A pattern is a $k \times k$ square of pixels in which certain pixels are required to have specific binary values. An example of a rule is shown in Fig.2. All occurrences of the lefthand side (LHS) pattern are simultaneously replaced by the righthand side (RHS) pattern. Since a pixel can be common to several shifted versions of the replacement pattern, the information in that pixel as a result of the replacement is a logical OR of the corresponding pixels.

Brenner, Huang and Streibl⁷ have suggested a set of substitution rules that are adequate to perform logical operations, thus demonstrating that SS is a general-purpose computing system. Murdocca¹⁵ proposed a general-purpose SS system that consists of only one substitution rule Fig.3. The choice of substitution rules is determined by such criteria as universality, simplicity, ease of implementation and efficiency. In particular, we show in Section 5.1.2 that the "complexity" of a rule influences the energy dissipation of the system.

A general-purpose computing system that employs SS has the following structure. The binary plane contains an encoding of the input data and control bits. The substitution rule will be applied to this plane repeatedly for a predetermined number of cycles. We can think of the control bits as the program. If we have several different rules, these can be applied serially or in parallel. When applied in parallel, the resultant plane would be the OR of the resultant planes from the individual substitution rules.

2.2.2. Optical Implementations of Symbolic Substitution Systems

An optical system for performing SS must provide two basic operations: pattern recognition and pattern replacement. The most widely used approaches for both operations apply a thresholding operation to a composite of shifted replicas of the input image. Here we will briefly review the ways that optical systems can produce shifted image replicas, and describe how this capability is combined with thresholding and logic-level restoration to provide cascable building blocks for pattern recognition and pattern substitution.

One important choice to be made in the specification of a pattern recognition module is whether it will recognize patterns of ones, patterns of zeros, or patterns consisting of ones and zeros. Recognition of patterns containing ones and zeros leads to system compactness and operational flexibility, but also requires greater complexity in the optical system.

2.2.2.1. Description of a Simple Recognition-Substitution Module

Implementation of SS is simplified if occurrences of a pattern consisting of only bright pixels (ones) or only dark pixels (zeros) is to be recognized. A bright pixel pattern recognizer will be described here. A replica of the input image is made for each bright pixel in the pattern to be recognized (the LHS pattern). Each replica image is shifted horizontally and vertically by an amount that brings a corresponding LHS bright pixel to the position of a designated origin pixel. All the shifted replicas of the input image are superimposed, producing a composite image having pixels with different brightnesses. The brightest pixels in the composite image will occur at each position where the input image matches the LHS pattern. The composite image is incident on an array of thresholding optical gates whose output leaves only these brightest pixels (pattern matches) in the bright state.

Once bright pixels marking the locations of pattern matches have been obtained, the next step is to substitute the new RHS pattern at each location. For each bright pixel in the RHS pattern, a replica of the image at the output of the threshold array is made. The replica images are shifted by an amount corresponding to the position of the bright pixels in the RHS pattern. The shifted replicas are superimposed (ORed), with the result that the RHS pattern now appears in all locations that a recognition spot existed. For achieving cascable modules, an array of gain and isolation devices is included.

2.2.2.2. Implementation of Image Shifting and Combining Operations

Optical implementations of symbolic substitution are all based on replicating, shifting, and recombining data page images. During pattern recognition, a shifted replica of the input image must be formed for each distinguished bit in the pattern to be recognized. For substitution, a shifted replica of the output of the threshold array must be produced for each bright pixel in the substituted pattern. Two approaches to replicating, shifting and combining images for symbolic substitution have been published in the literature:^{7,14} geometrical optics using beam-splitters, mirrors, and prisms; and diffractive optics using holograms. We will briefly review the merits and fundamental limitations of each in the following:

Several systems using geometrical optics components have been proposed for providing the image replication, shifting and combining operations for symbolic substitution. All of them are roughly equivalent to the beam-splitter configuration shown in Fig.4. Though these implementations are very straightforward, the process is inherently power inefficient. In principle, two images may be combined without power loss with the use of a polarization beam-splitter, but the output image, containing both polarizations, is not suitable for cascaded stages of lossless combinations. However, many rules require detection and substitution of patterns containing at least four or more shifted images. Thus, a spatial light modulator must be used to regenerate an image with one linear polarization after each pair combination, or non-polarized image combination must be used for each additional image combination. If this second approach is adopted, then to combine N images, at least $(N/2-1)/(N/2)$ of the input power is lost.

The alternative to geometrical optics for image replication, shifting and combining is the use of holograms. In contrast to geometrical optics, volume holograms can, in theory, be used to losslessly combine many images. A more subtle problem arises with the use of holographic optical elements (HOE's) however. Holograms do not delay wavefronts the way refractive optical components do. With holograms, all phase delays are modulo two π . This means, for instance, if a hologram performs the function of a lens, wavefronts passing through the center of the holographic lens will arrive at the image before those passing through the edge. Put another way, pulses of light will be stretched in time, placing a lower limit on the clock period for an optical system. As an example, a 2.5cm diameter $F/1$ holographic lens will lengthen all pulses of light passing through its full aperture by about 50 psec.

2.2.2.3. Data Encoding Schemes

Two approaches have emerged for recognizing patterns containing both ones and zeros. The first approach is dual-rail logic, or position encoding. With this method both the true and false states of a logic variable are represented by a bright spot in the optical array; ones are represented by a bright spot in a specified position, zeros by a bright spot in another position (e.g., see Fig. 2). Thus, the problem of detecting ones and zeros in a pattern has been translated into a requirement to detect just ones or just zeros. Processing can proceed as described before for those operations.

The other approach to detection of patterns containing both ones and zeros is to encode the binary states of a cell not with intensity but with orthogonal polarizations of light.¹⁵ As with simple recognition, a replica of the data plane is produced for each distinguished cell in the LHS pattern, but in this case both true and false LHS cells may be specified. Replicas corresponding to zeros in the LHS pattern are passed through a half-wave plate, thereby inverting the logic value of their bits. Shifting now occurs on all replicas to bring the specified LHS cells to the origin. The resulting superposition passes through a polarizer aligned with the true state polarization in the data array.

Wherever the data array matches the LHS pattern, all cells with the true state polarization and cells with a false state polarization that has been rotated 90° to the true state are superimposed. Thus matches are noted by the brightest pixels after passing through the analyzer. From this point, the rest of the process follows that for simple recognition.

Both of these approaches roughly double the power consumed by the system. For dual rail encoding this occurs because the number of pixels to represent each bit is doubled, and the complexity of logic and data paths is correspondingly increased. For polarization-based encoding, a polarization analyzer is used prior to the optical gates, thereby discarding half of the power.

3. Symbolic Substitution Systems and 2-D VLSI Mesh

In this section, we compare optical SS systems with a VLSI two-dimensional mesh of processors that operates in SIMD mode. We will show that a SS system can be efficiently simulated by a very fine-grain mesh of processors, and that a SS rule can be simulated using only a small number of cycles that depends on the size of the SS rule. In fact, we specify measures to quantify the complexity of a SS rule. On the other hand, we show that a SS system is inefficient in simulating a mesh of electronic processors, where each processor has the ability to perform basic arithmetic and data movement operations on 8-bit words. This simulation requires more space and time. We also give quantitative estimates of the resources needed to simulate an optical symbolic substitution system using a mesh of VLSI processors. As in a mesh, in SS each instance of the rule works only on a small amount of nearby information.

3.1. Simulation of SS by a Mesh

We make first the following assumptions about the mesh. Each processor is connected to its four nearest neighbors with bidirectional edges. The operation of each of the processors is synchronized by a global clock. Each processor has instructions for communicating with its four neighbors and for computing the logical operations AND, OR and NOT.

To simulate an $N \times N$ optical symbolic substitution system by an $N \times N$ mesh of electronic processors, further assume, without loss of generality, a symbolic substitution system based on a single rule, then extend our analysis later to handle the case of multiple rules. The basic idea is that each mesh processor (x,y) is responsible for the state of the pixel (x,y) in the binary plane of the symbolic substitution system. We then simulate the transition rule on the mesh and update the states. In the following, we compute the cost of simulating a transition rule.

Consider a transition rule that replaces a $k \times k$ frame with another $k \times k$ replacement frame based on the existence of a certain search pattern in the frame. A search pattern is specified by requiring *distinguished* pixels, to have certain states. Let m be the number of these pixels. The other $(k^2 - m)$ pixels in the frame are don't-care pixels because their state does not affect the recognition of the pattern. Similarly, the replacement frame is specified by giving the set of *distinguished* pixels that are required to have the value 1. Let n be the number of those pixels. Other pixels in the replacement pattern have the value 0. Our aim is to capture the cost of the complexity of simulating the transition rule as a function of the k, m and n .

Consider how a pixel in the output plane of a symbolic substitution system can possibly change its state after an application of a transition rule. Each pixel in the output plane depends on exactly n $k \times k$ frames. If, at least one of these frames have the required search pattern, a 1 will be written in the pixel. The presence of a search pattern in a frame is determined by the m distinguished pixels.

Hence, the new state of a cell is determined by a Boolean formula which is an OR of n terms each of which is an AND of m Boolean variables. We will next show how this function can be computed for each of the pixels in parallel in time $O(k^2)$ and with a small ($O(\min(n, 2k))$) amount of hardware per processor in the mesh.

In the first phase, we compute the AND of the distinguished pixels for each possible $k \times k$ frame. For each frame, we designate a unique pixel to collect and AND together the states of the distinguished pixels corresponding to the search pattern. Note that each pixel appears distinguished in the search patterns of exactly m frames. Hence, each pixel has to send its state to m different recipients. This transmission can be accomplished in $(k-1) + \min((k-1)m, k(k-1))$ communication cycles of the mesh. Furthermore, each processor in the mesh need only have $O(\min(m, 2k) + \log k)$ switches. At the end of the first phase, all the required products are computed. In the second phase, each distinguished pixel of the replacement pattern receives n of these products and computes their OR. This again can be accomplished in $(k-1) + \min((k-1)n, k(k-1))$ communication cycles of the mesh with at most $O(\min(n, 2k) + \log k)$ switches per processor. In summary, a transition rule can be simulated on the mesh in time $O(\min((m+n)k, k^2))$ with $O(\log k + \min(m+n, k))$ switches per processor.

These bounds work in general. In many specific cases, one could exploit the regularity of the rule to derive more efficient simulations. For example, Murdocca's transition rule can be simulated in about 8 communication cycles.

The simulation procedure described above does not handle the processors at the edges nor the case of a system where several rules are being applied simultaneously. The edge processors can be taken care of by deleting the appropriate product terms. We can simulate a system with several rules by considering the logical OR of the output binary planes from applying the individual transition rules. The cost functions for this case would be the same as in the one-rule case with $k = \max(k_i)$, $m = \sum m_i$ and $n = \sum n_i$.

Since there is a limitation on the size of electronic mesh that can be implemented presently, we should consider the problem of simulating an N by N symbolic substitution system with a smaller M by M mesh, where $M < N$. We assume N/M to be some integer multiple of k , and compute the time and space requirements to perform this simulation. The basic idea is to make each processor in the mesh responsible for a p by p window of pixels in symbolic substitution binary plane, where p is N/M . The simulation algorithm we use here is composed basically of a communication phase followed by a computation phase. In the communication phase, each processor sends pixel state information to its four nearest neighbours such that $4p(k-1) + 4(k-1)^2$ state bits are received at each processor. The idea is that each processor gathers a $k-1$ wide window of states around it so that it has all the necessary information to compute the new states of its pixels. The time for the computation is $O(p^2 \log(mn))$ and each processor needs $O(mn + p^2 + 4p(k-1) + 4(k-1)^2)$ switches. The overall time for simulating one application of a transition rule is $O(p^2 \log(mn) + 4p(k-1) + 4(k-1)^2)$. In particular, when $p \gg k$, the time is $O(p^2 \log(mn))$ and the hardware cost is $O(p^2)$.

3.2. Simulation of a Mesh by SS

We now consider the simulation of a VLSI mesh with a SS system. It will be shown that such simulation requires more space and processing cycles, even for a very simple mesh.

Consider a mesh of 1-bit processors, each having three registers capable of performing logical and data movement operations. We have also instructions to transport the data between the

neighboring processors. To simulate such a system, we make the following two generous assumptions about the capabilities of the SS system. i) The system can have a large number of substitution rules operating in parallel. ii) The control bits in the input plane can be changed every cycle.

The basic idea of the simulation is to allocate a window of SS pixels for each processor. This window contains the space for the three registers and the control bits to specify the instruction in dual rail logic. We use multiple SS rules (about 16) operating in parallel to implement the instruction set.

This scheme gives us the minimal area per processor and 1 cycle time to execute an instruction. Simple calculations show that the area required per processor would be at least 25 pixels. Thus, if we assume that the binary plane has 1000 x 1000 pixels, we can at best simulate a 200 x 200 mesh of 1-bit processors with each step of mesh taking 1 clock cycle of the symbolic substitution system.

If a larger-grain processor is used or if the above-mentioned assumptions are not feasible, in particular if we have to work with a single rule, then the corresponding simulation would be much more inefficient both in terms of time and area. This would imply that any realistic SS system can only simulate a small mesh (less than 100 processors) taking a large number of cycles to simulate a cycle of the mesh.

To summarize, we have shown that a SS system is no more powerful than fine-grain mesh of processors of similar size. This means that any advantage that can be enjoyed by an SS system must come from technological considerations. In the next section, we look at the technological aspects.

4. System and Technological Considerations of SS and POEMs

Here we discuss the technological characteristics of both SS and POEMs systems. In particular, we determine the energy dissipation and speed of these systems. To begin with let us consider some fundamental characteristics associated with optical gates these systems are made of.

4.1. Fundamental Considerations for Optical Gate Arrays

In the following, we analyze optical gate switching speed and array size in terms of thermal limitations, optical interconnect density and efficiency of optical and electrical interconnects.

In general, a bound on the number ($N \times N$) of gates in an array of area A can be found by requiring that heat dissipation cannot be larger than the heat removal per switching cycle. Thus, we have

$$N^2 \leq \frac{P_{dmax} A}{P_c A_c}, \quad (1)$$

where P_{dmax} is the maximum allowable power dissipation density, which is dependent upon the thermal characteristics of the material and the heat removal technique applied to the device. P_c is the power dissipation density of a single optical gate and A_c is its active area. In addition, the required space bandwidth product SBP of an optically interconnected system will be

$$SBP \geq \frac{A}{A_c}. \quad (2)$$

In general, A is limited by wafer size and A_c is limited by lithography or by the optical wavelength. Combining equations (1) and (2) we obtain an upper limit on the size of an optical gate array

imposed by thermal dissipation and optical interconnect density as

$$N^2 \leq \left[\frac{P_{dmax}}{P_c} \right] SBP. \quad (3)$$

For an optical gate, the power dissipation density is related to the switching-energy density E_c and the switching speed τ by $P_c = E_c / \tau$. Using this relation in eq.(3), one can show that the minimum switching speed τ of the array will be determined by

$$\tau = \frac{E_c}{P_{dmax}} \left[\frac{N^2}{SBP} \right]. \quad (4)$$

Hence, for a given device and optical interconnect technology, the speed of an optical gate is limited by the array size. An important figure of merit for optical gate arrays therefore is the array throughput given by

$$N^2 \tau^{-1} \leq SBP \left[\frac{P_{dmax}}{E_c} \right]. \quad (5)$$

This equation puts an upper limit on the capabilities of any optical gate array implemented with a given technology. In the case of the opto-electronic PE arrays used in POEMs, A_c is the area of a single modulator in each PE and is occupying only a small fraction of the total PE area. With the simplifying assumptions made in section 5.1 for the worst case calculations, equation (5) can also be used to estimate the computational throughput of POEMs. Next, we develop models to evaluate the energy dissipation and the latency of POEMs and symbolic substitution systems.

4.2. Energy Dissipation and Latency for Symbolic Substitution and POEMs

In this section we determine the energy dissipation and the speed of POEMs and symbolic substitution systems.

4.2.1. POEMs

The POEMs machine is composed of electronic processing elements (PEs) interconnected with holographic-optical interconnects. Each PE is made of logic gates interconnected with electrical interconnects. The energy is dissipated essentially in the electrical interconnections and in silicon inverters. The maximum PE clock rate is fundamentally determined by the speed of the longest electrical interconnect in the PE, while the speed of interprocessor communication is determined by the longest holographic interconnect in the system.

First we discuss the energy dissipation and the speed of a PE. The total energy dissipated per clock cycle within a processing element is the sum of the energies spent in switching the electronic logic gates and driving the interconnects. The energy consumed in switching a logic gate with short connections is dominated by the gate input capacitance (C). If V is the required voltage swing, then the switching energy is given by

$$E_c = \frac{CV^2}{2}. \quad (6)$$

When the connections are longer, the wire capacitance dominates the gate input capacitance, and the switching energy becomes proportional to the length of the electrical wire.

The operating speed of the circuit is inversely proportional to the connection delay, which depends on the length of the wire. For short wires it is given by¹⁶

$$\tau_{\text{short wire}} = 2.718 \tau_{\text{inv}} \ln(K L_{\text{wire}}), \quad (7)$$

where L_{wire} is the connection length, and K is a constant, typically between 0.1 and $0.2 \mu\text{m}^{-1}$. The inverter switching time (τ_{inv}) is a technological constant representing the logical gate switching speed. This logarithmic dependence of wire delay on wire length shows that, for locally connected gates, the speed is essentially determined by τ_{inv} . On the other hand, when the connections are long the wire delay is proportional to the wire length and is given by¹⁶

$$\tau_{\text{long wire}} \geq \frac{L_{\text{wire}} \sqrt{\epsilon_r}}{c} \quad (8)$$

where c is the speed of light and ϵ_r is a constant, typically about 4. Thus, long electrical connections decrease the speed of operation and increase the energy consumption.

We now turn our attention to the energy dissipation and speed of holographic interprocessor connections. Figure 5 illustrates a free space optical interconnect system. A biasing optical field is incident only on the modulators associated with each PE. The light, transmitted by a modulator that is turned on, is directed with holographic interconnect onto the desired detector(s). The energy required by such interconnects can be evaluated to be¹¹

$$E_o = 2VF(C_{\text{pd}} + C_{\text{inv}}) \times (2h\nu/(\eta q) + V) + C_M V_M^2, \quad (9)$$

where E_o is the required optical link energy, V is the inverter voltage swing, F is the fanout, C_{pd} is the photodetector capacitance, C_M is the modulator capacitance, and V_M is the half-wave voltage of the modulator. The photon energy is represented by $h\nu$, and the electronic charge by q . The efficiency of the optical link is modeled by η , which includes the efficiencies of the modulator, the hologram, and the detector. When compared with the energy requirements of electrical interconnects in eq. (6), it can be shown that E_o is less for long communication distances. The break-even communication length establishes the criteria for the appropriate use of electrical and optical interconnections. As an example, an optical link realized with a PLZT light modulator with $10 \mu\text{m}^2$ area and a fanout of 1 using the $2.5 \mu\text{m}$ process will dissipate an energy of 50pJ, assuming 60% holographic diffraction efficiency, and 90% modulator and detector efficiencies. When compared to the energy required for a typical electrical off-chip connection of about 1nJ, the optical link consumes less energy.

The speed of operation of POEMs can be limited by the latency of the global optical links, local electrical interconnect delay, or the inverter switching speed. The latency of the global optical links will be governed technologically by the light-modulator speed and fundamentally by the skew introduced by holographic interconnects and by the free space optical propagation delays. Typical achievable speeds with light modulators are 0.1 to 1 microseconds with Si/PLZT and 1 to 10 nanoseconds with MQW technologies.¹⁷ For global holographic optical interconnections relative time delays will be introduced among the PE's by the hologram. This skew can be expressed from simple geometrical considerations as

$$\tau_h = (f/c) \left[\sqrt{1 + \frac{D^2}{f^2}} - 1 \right], \quad (10)$$

where f is the distance between the PE array and the hologram and D is the length of a side of the array. For fixed interconnects this skew can be compensated by the introduction of appropriate

optical time delay elements into different communication paths. However, in the case of programmable optical interconnects, this compensation technique cannot be used because of the time dependence of the relative delays. Nevertheless, the magnitude of the skew is presently less than the latency of the state of the art MQW light modulators and therefore does not limit the communication speed. For example, for an opto-electronic PE array 15 cm on the side, the signal skew ranges from 20ps to 200ps as l is varied from 150 to 15 cm. Note that the magnitude of the skew is reduced by increasing l . However, the free space optical propagation delay increases with l according to $\tau_{tr} = l/c$. Thus for a given array dimension there exist an optimal distance l , minimizing the propagation delay and signal skew.

4.2.2. Symbolic Substitution

Here we compute the energy dissipation and the delay involved in a single application of a symbolic substitution transition rule. Figure 6 shows the diagram of the system we use. We make the following assumptions about the system:

- i. The symbolic substitution system uses a single rule of (k,m,n) complexity operating on an $N \times N$ pixel image.
- ii. The input image contains b bright pixels, each having an energy of e_{in} .
- iii. The input image contains S occurrences of the search pattern.
- iv. The transition rule produces an $N \times N$ output image with S' bright pixels, each having an energy of e_{in} .
- v. The optical operations of splitting, shifting, combining and imaging are lossless.
- vi. The system contains two $N \times N$ arrays of optical gates: one for logic-level isolation and restoration (amplification), and the other one for thresholding.
- vii. Figure 7 shows the model of the three-terminal device used in the amplifier array. When light of energy e_{in} enters the device, light of energy Ge_{in} leaves. In this case, conservation of energy requires that

$$e_b + e_{in} = Ge_{in} + e_{ca}, \quad (11)$$

where e_b is the bias energy to the amplifier, G is the input-output gain of the device, and e_{ca} is the energy dissipated in switching the device (Fig. 7a). On the other hand, when no input light enters the device, the bias energy of e_b is dissipated by the device and no output is produced (Fig. 7b).

- viii. Figure 8 shows the model of the two-terminal device used in the threshold array. Such a device is characterized by its threshold energy and switching energy e_{ct} . When the input light is below the threshold energy, no output is produced and all of the input energy is dissipated at the device. But, when the input light energy exceeds the threshold, output light of energy $(e_{in} - e_{ct})$ is produced and an energy of e_{ct} is dissipated in switching the device.
- ix. The devices are memoryless, that is, at the end of each clock cycle the optical gate-arrays are reset.

We now explain the system energy budget shown in Fig. 6. The bias energy of $N^2 e_b$ is used to power the amplifier array. Since the input has b bright pixels, the output of the amplifier array also

has b bright pixels, each having energy of $G e_{in}$. The total energy dissipated in the amplifier array is the sum of the energies dissipated by the b switching cells which had light incident on them ($b e_{ca}$), and the $N^2 - b$ cells which had no light incident on them ($(N^2 - b) e_b$).

The amplifier array is followed by a lossless optical system that produces an image with S pixels above the threshold energy of the threshold devices. This image is incident onto the threshold array. The energy dissipation in the threshold array is the sum of the energy to switch the devices for S pixels above threshold ($S e_{ct}$) and all the energy that is below threshold ($(b - S) G e_{in}$). The output produced by the threshold array is an image with S bright pixels each having an energy of $(G e_{in} - e_{ct})$. This image is passed to the optical system for substitution which generates a final output image with S' bright pixels.

Conservation of energy requires that the total energy entering the system is equal to the energy leaving the system plus the energy absorbed. Using this constraint we obtain

$$e_b \geq (n - 1) e_{in} + e_{ca} + e_{ct}. \quad (12)$$

This equation reveals that each pixel needs enough energy to create a full rule-substitution pattern and to energize one threshold device and one amplifier device. Now we use eq. (11) to eliminate the dependence on e_{in} in eq. (12) to obtain:

$$e_b \geq e_{ca} + \left[\frac{G - 1}{G - n} \right] e_{ct}. \quad (13)$$

This equation indicates that the gain of each amplifying device must exceed n . The overall energy dissipation can now be computed by adding the energy dissipations of the amplifier and the thresholding arrays and using eq. (13) for e_b . This gives:

$$E_{diss} = N^2 e_b + N^2 \left[\frac{b - S'}{G - n} \right] e_{ct} = N^2 \left[\frac{G - 1}{G - n} \right] e_{ct} + N^2 e_{ca} + N^2 \left[\frac{b - S'}{G - n} \right] e_{ct}. \quad (14)$$

The first term in eq. (14) is the energy required to bias an array of $N \times N$ optical devices such that the recognition-substitution operation can be carried out. This amount of energy is dissipated under all conditions. According to eq. (14), the bias energy is quite large because it is $N^2(G - 1)/(G - n)$ times the switching energy required per thresholding device plus N^2 times the switching energy required by the amplifying devices. The second term in eq. (14) represents the energy losses associated with different fan-in and fan-out and can be made to vanish for $m=n$, i. e. for constant fan-in and fan-out. For example, assuming G of 5 and using Murdocca's simplest rule where $(k, m, n) = (3, 4, 4)$, we require power dissipation of $(36 e_{ct} + 9 e_{ca})$ for each 3×3 window. Considering that the switching energy of an optical device is presently equal to the energy of an electronic transistor, Murdocca's symbolic substitution rule requires the energy equivalent of 45 transistors. We shall discuss the computational value of such a recognition-substitution module in the next section.

The above argument can easily be extended to symbolic substitution system with R parallel rules. Such a system is basically equivalent to R one-rule symbolic-substitution systems operating in parallel. Assuming the same gain for the amplifying array, the energy dissipation will be essentially R times larger and is given by:

$$E_{diss} = N^2 \left[\frac{R(G - 1) + (b - S')}{(G - 1) - R(n - 1)} \right] e_{ct} + N^2 e_{ca}. \quad (15)$$

We now estimate the latency of a recognition-substitution module. The time required to perform one application of a symbolic substitution rule is the sum of the time required to switch the optical gates and the transit time through the optical system. The speed of the optical gate is limited by the array size and is given in eq. (4). The transit time is limited by the complexity of the rule and the imaging optics. For example, for a very simple symbolic substitution rule such as the one proposed by Murdocca, the transit time (τ_{transit}) is proportional to

$$\tau_{\text{transit}} = \frac{4f}{c}, \quad (16)$$

where f is the lens focal length. Using the expression for resolution of an Airy pattern we can express the latency in terms of the SBP of the optical system, the F-number of the lenses ($F_{\#}$) and the optical frequency (ν) as

$$\tau_{\text{transit}} \geq 10\nu^{-1} \sqrt{\text{SBP}} (F_{\#})^2 \quad (17)$$

or using equation (3)

$$\tau_{\text{transit}} \geq 10\nu^{-1} (F_{\#})^2 N \sqrt{\frac{P_c}{P_{\text{dmax}}}}. \quad (18)$$

Note that the latency of a SS system grows as the side of the array. It should be noted that for more complex systems, the optical transit time increases with the parameter m of the rule and the number of rules R .

5. Relative Merits of SS and POEMs

Based on the previous analysis, we now compare quantitatively the performance potential of symbolic substitution and POEMs systems.

5.1. Computational Efficiency

The computational power of symbolic substitution systems lies in their ability to implement space-invariant transition rules very quickly. The communication involved in effecting these transition rules is done by replicating, shifting and combining images. Such operations are easy to accomplish in optics.

But this capability of symbolic substitution systems does not necessarily translate into computational efficiency. The computation involved is done by thresholding or clipping (a non binary operation) an analog signal back to binary form, resulting in inefficient energy utilization, especially for complex rules. The communication provided by the transition rules is very local and space-invariant. However, many computations, including basic operations like addition and multiplication, can be implemented more efficiently with space-variant communication. Hence, SS requires a large number of pixels and substitution cycles to implement operations like logic functions, addition, multiplication, etc.

To provide a specific example, consider the implementation of a NAND gate using Murdocca's rule¹³. The symbolic substitution NAND gate takes 255 pixels of area and requires 6 applications of the transition rule. In contrast, an electronic NAND gate requires 4 inverters and takes about one inverter switching time when short wires are used. That is, a NAND gate fabricated with 1 micron CMOS lithography that has a fanout of two takes 400 picoseconds when the connection length is less than 1 mm. If the inverter switching energy and the optical gate switching

energies are assumed to be the same, then Murdocca's NAND gate requires four orders of magnitude more energy than the electronic NAND gate. Another example of wasted space is shown in Fig. 9 where a Flip-Flop is implemented with 50x56 pixels using Murdocca Rule. Additional examples given by Cloonan^[23] and Goodman^[24] show that many other important Boolean logic modules require more area and time when implemented with symbolic substitution.

In section 3. we had shown that it requires a large area to implement a basic processor capable of arithmetic and data movement operations. In section 4. we have derived the limitation in size and speed of optical gate arrays in terms of thermal considerations. In the following, we show how power considerations limit the size and speed of POEMs and SS systems. In particular, we derive some estimates on speed and size based on the value E_c , the minimal device switching-energy density. We show that even the best possible values of E_c cannot support a large and fast SS system.

5.1.1. Speed and Size

The speed and size of both systems are governed by equation (5). With respect to SBP, POEMs may enjoy three orders of magnitude advantages over SS, since the POEMs machines use diffractive optics for global connections, while in symbolic substitution all interconnects are implemented with refractive optics. Using multilevel-phase holograms, the SBP of diffractive interconnects can be as large as 10^{11} . On the other hand, lens-based refractive interconnects have a SBP of at most 10^8 . The large SBP of holographic interconnects is used in the POEMs architecture to achieve a larger ratio A/A_c in equation (2) while retaining a high degree of concurrency and allowing reasonable area to implement electronic signal processing.

We now consider the information handling capacity of the POEMs machines for two different opto-electronic technologies: Si/PLZT⁹ and Si or GaAs IC integrated with Multiple Quantum Well (MQW) modulators.¹⁷ We assume that a processing element with all required operations incorporated can be implemented in a square area of $10^5 \mu\text{m}^2$ using 0.5 micron CMOS technology. This number is calculated based on the layout of a prototype PE designed with 2.5 μm minimum feature in 1 mm^2 area. We also assume that the size of the processor plane is limited to 6"x6" by the wafer size. Then, there will be 250,000 PEs on the processor plane. Given a maximum power dissipation density of 10 watts/ cm^2 , we perform our calculations for the worst case conditions assuming that all devices on the wafer dissipate the same switching energy that is required to drive the optical devices. Si/PLZT technology requires 1 pJ/ μm^2 switching energy density for the PLZT modulators and a typical modulator occupies 10 μm^2 area. Using equation (5) we obtain $N^2 \tau^{-1}$ of 10^{16} operations/second. If we wish to achieve a throughput of 10^{12} ops/sec, a silicon area equally to that taken by 10^4 optical modulators ($10^4 \times 10 \mu\text{m}^2$) can be used effectively to host one PE. Thus, with a 100% yield on a 6 inch wafer, 250,000 globally interconnected PEs, each containing roughly 1000 gates, can be operated at Megahertz rates.

Assuming a Si/MQW or GaAs/MQW integration technology to be available, a similar calculation reveals that one can implement 250,000 PEs, all communicating with one another at 0.1-1GHz rate, because $E_c = 10 \text{ fJ}/\mu\text{m}^2$, which is smaller than $E_c = 1 \text{ pJ}/\mu\text{m}^2$ for Si/PLZT. Note that the fundamental limits on the speed achievable with POEMs is limited to a few gigahertz by the skew associated with the optical transit time in the global holographic interconnects. It should be noted also that the processing elements can perform local computations at rates higher than the communication speed.

Holographic interconnects cannot be used in symbolic substitution because of the pulse spreading they introduce at very high speed operation. Therefore, refractive interconnects must be used in symbolic substitution, limiting SBP to 10^8 . As can be seen from eq. (4), using MQW technology with $N^2 \tau^{-1} < 10^{15}$, only 1 million optical devices will be allowed to operate at a maximum switching rate of 1GHz. Devices under development that require $E_c = 1$ fJ/micron² at 10 picosecond switching rate will allow a maximum optical gate array size of 316x316. In addition to the device size limitations, systems using such high speed devices will be limited by the optical transit time as given by equation (18).

5.1.2. Complex and Multiple Rules

One can alleviate some of the speed and size inefficiencies that accompany a single rule system such as Murdocca's by using complex and/or multiple rules operating in parallel.^{8,18} However, complex and multiple rules increase the energy consumption and decrease the speed of the system, as shown in eq. (15). There are additional constraints on the complexity and the total number of rules in a symbolic substitution system. For example, the m parameter of a symbolic substitution rule cannot exceed the dynamic range of the thresholding device. For proper recognition, the thresholding devices must be able to distinguish an input light intensity of e_{in} from $(1 - \frac{1}{m})e_{in}$. The n parameter is limited by the gain of the amplifier device. As seen in equation (13), n imposes a lower limit on G . Increasing the number of rules in the system increases the optical transit time, the size of the system, and dissipates more energy. For large number of rules (R), the increase in energy dissipation is by a factor of R as can be seen from eq.(15). The optical transit time increases as we stack many images of the binary plane in free space. These multiple images also increase the total volume of the system.

Based on the above discussions, complex/multiple rules are not favored in an SS system. As a consequence, we cannot easily eliminate the size and speed inefficiencies that accompany the implementation of basic operations by symbolic substitution rules.

5.2. Architectural Considerations

Since technological considerations do not favor highly complex rules, the communication in a symbolic substitution system is essentially local. In section 3, we showed that architecturally a symbolic substitution system is not more powerful than a mesh. In this section, we argue that the mesh architecture is not always an efficient network topology for parallel computation even though it is easy to implement. We can map certain problems efficiently onto a mesh by using highly regular algorithms, consequently facilitating very fast communication. But, these highly local interconnections limit the performance of many algorithms. Any algorithm whose output depends on almost all of the inputs requires at least N , time steps, the diameter of the mesh. On the other hand, networks like the hypercube have $\log(N)$ diameter. Even though the communication in these cube-like architectures tend to be slower than that of the mesh, diameter consideration indicates that these networks will ultimately be more efficient. Table I shows that several important prototype problems do have more efficient algorithms on highly interconnected architectures. Hence, the real question is whether better communication schemes can be developed for the cube-like architectures and at what point it is advantageous to have slower communicating but more globally connected processors.

POEMs offer a potential solution because their architecture overcomes many of the limitations that highly interconnected electronic architectures like hypercube face. POEMs provides a flexible, fast and parallel environment through its programmable global optical interconnects.

First, POEMs can handle half a million PEs on two highly interconnected wafers, as discussed in section 5.1. This large number and very high density of interconnections is a direct result of the 3-D nature of the POEMs architecture. Although the estimated number of PEs in POEMs is already quite impressive, one can envision that the number can be further increased using more PE planes and interconnection holograms. Additionally, with multiple PE planes the processor grain size can be increased without reducing the overall number of processors in the system.

Second, optical interconnects provide fast means of global communication with low energy requirements. As a result, POEMs can fully use the advantage of highly interconnected architectures.

Third, the topology of the interconnects in POEMs is not restricted to being regular. Space variant interconnection holograms^{19,20} allow arbitrary and irregular communication between processors. In fact the need for such communication is supported by the theory of parallel algorithms which shows that fast parallel algorithms require irregular communication among PEs.^{21,22}

Finally, POEMs architecture allows for programmable interconnections. This reduces the silicon area required by the routers commonly used in electronic concurrent computers. In addition, such programmable interconnections are desirable, since different algorithms dictate different interconnections for efficient implementation.

5.3. Other Considerations

In the following subsections we compare local communication, programming methodologies, and the resistance to technological defects in symbolic substitution and in POEMs.

5.3.1. Local Communication

In this subsection we show that "simulated wires" used in symbolic substitution are much slower than electrical wires used in POEMs. The idea of 'simulated wires' is repeated application of a rule to move information across the plane. In fact, one application of a rule of complexity (k,m,n) can move information by a distance of at most (2k-1) pixels. If τ is the time required to apply the rule, and L is the length of the connection in pixels, then the

$$\tau_{\text{wire}} = \tau \frac{L}{(2k-1)}. \quad (19)$$

In particular, for Murdocca's rule with k=3, the movement of data across a distance equal to the length of several gates requires hundreds of applications of the rule because even the simplest logic gates require large area when implemented in symbolic substitution.

In contrast, the delay of a short electrical wire is basically determined by the inverter switching time and is given in eq. (7). Moving information, over a short distance or across many hundreds of logic gates, takes essentially the same amount of time. This is due to the small size of electronic logic gates and the logarithmic dependence of delay on the length of the wire.

To illustrate the above arguments we compare the time delay in moving information in symbolic substitution and in POEMs. Assuming that the time to apply one transition rule is 1 nanosecond and the area of a typical logic gate is 10 x 10 pixels, the time to move information across

10 gates is 100 nanoseconds. In contrast, a NAND gate, in 1 micron CMOS technology, with a 1mm long output wire, has a delay of about 400 picoseconds and can move information across many hundreds of gates.

In summary, the 'simulated wires' of symbolic substitution are inferior to their electronic counterparts in speed. This result indicates that the data and the control bits that operate on it must be placed close together in the plane.

5.3.2. RAM implementation and Programming Methodologies

The programming flexibility of digital electronic computing is closely associated with its ability to implement random access memories(RAM). In this subsection we show that symbolic substitution does not provide an efficient means of RAM implementation, limiting its applications and increasing programming complexity.

In electronics, the speed of local interconnects enable the implementation of small size RAMs with fast access times. This enables POEMs machines to perform space-time trade-offs and to handle large problems. In particular, it enables POEMs machines to perform context switching for solving problems larger than the size of the machine. Therefore, POEMs programming can be accomplished using the conventional stored program concept. The instructions and the data can be stored in the memory and executed by a processor.

On the other hand, symbolic substitution has slow local communication making the implementation of RAM difficult. A RAM has a requirement that any bit of storage is accessible in one clock cycle. Consider an S^2 bit symbolic substitution RAM. Assuming that the RAM is laid out as a 2-d array of $p \times p$ pixel windows and each window stores one bit, the length of the side of the array is $S \cdot p$. Thus, the longest simulated wire in this system is about $S \cdot p$ pixels long. Implementing even 100 bits of memory with Murdocca's rule or another similar rule would require unacceptable access time due to wire delays. Thus, the programming methodology in symbolic substitution must be different from the processor-memory model used in POEMs and at least for now appears to be more difficult and limited in flexibility.

To overcome the lack of efficient communication capability, efficient memory and complex logic implementations, researchers have proposed to lay out symbolic substitution programs as circuits, with data and associated control bits being placed in close proximity. This approach seems to be harder, at least for now, to use for programming because of the difficulty of laying out the computation and making sure that the timing is properly arranged. Thus, it appears that symbolic substitution would be more suited for highly structured, local, fine-grain, space-invariant problems. An application of symbolic substitution to such a problem has yet to be demonstrated.

5.3.3. Fault Tolerance

Any fabrication procedure has a certain yield factor. Therefore, the POEMs and SS systems must have resistancy to technological defects. In POEMs architecture the global interconnects are space-variant and programmable. Therefore, faulty processors can be easily bypassed.

Since the interconnections in symbolic substitution are space invariant and non-programmable, it seems very difficult to implement any fault tolerance. One possible way of handling faults is to arrange the control and data bit placements to avoid defective cells. Although this is possible, it complicates the design of SS systems because it introduces additional constraints to the problem of laying out a computation in the SS plane.

6. Conclusions

Our intent in this paper has been to introduce a new optoelectronic parallel computing architecture called POEMs. Further, the attractive features of this architecture have been established by comparing POEMs to symbolic substitution, a parallel optical computing system widely recognized in the research community. The comparison has included computational efficiency of the architectures, power dissipation and speed of the respective supporting technologies, ease of programming, and amenability to fault tolerance. A summary of the comparison appears in Table II.

The POEMs architecture is motivated by analyses indicating efficient and effective means of combining optics and electronics. Electronics possesses a very mature technology for switching devices, and electrical communication is more efficient than optical for short distances (less than 1 mm). Thus, small to medium grain electronic processors (about 1000 gates) form the core of POEMs. For the greater distances of interprocessor communication, optical link efficiency compares so favorably with electrical that the price paid (in power dissipation and delay) for optoelectronic conversions is overcome. In contrast, symbolic substitution is at the extreme of fine grained processing and pays a high price for having even its shortest links implemented optically. All-electronic multiprocessor systems usually represent the other extreme of coarse-grained processing and squander substantial power and time by driving long wires.

Also, POEMs can incorporate complex global patterns of interprocessor communication which are difficult for symbolic substitution and all-electronic systems to achieve. The extremely high clock rate of SS systems prohibits the use of holographic connection elements, which in turn has required space-invariant communication using refractive optics. This reduces SS to the equivalent of a 2-D mesh-connected architecture, which is well known to be computationally inefficient for the solution of many problems. Though the use of multiple and complex substitution rules mediates against this limitation, such rules exact a heavy penalty in system power dissipation and speed. In this respect, our results agree with those of other researchers^[23] who show that space-invariant transition rules do not give efficient implementations of basic computational operations. Thus, we observe that some proponents of SS and similar systems have begun incorporating global interconnections into their designs^[25,26].

Although the technology supporting POEMs is not as fast as that used in SS, its efficient combination of optics and electronics and its flexible use of global interconnects gives POEMs computational power greater than that of SS. Efficient local electronic connections used in POEMs allow easy implementation of random access memory. This in turn facilitates traditional programming methodologies. In SS, data and programming information must be tightly interleaved because communication distances are limited. The space-variant and programmable optical interconnects of POEMs allow interprocessor connection topologies that are more efficient than mesh connection and easily accommodate fault tolerance through bypassing defective processors. The space-invariant connections of SS make this difficult to do.

Though the computational performance of POEMs using existing technology is already competitive with any other system, it can be expected to improve steadily. Current limitations to POEMs performance are technological: the speed of electronic processors and optical modulators, both of which are being actively developed. SS, by relying heavily on high speed for computational power, already faces fundamental limits in device power dissipation and signal skew due to optical propagation.

POEMs is well suited for parallel processing with a variety of processor granularity, synchrony, and interconnection topology. It combines the power of parallel space-variant optical

communication with the flexibility and efficiency of electronics. The fast, global, and programmable interconnections of POEMs will enhance significantly the capabilities and application range of parallel computing.

Acknowledgment

This work has been funded by DARPA under the management of AFOSR by contract 88-0022 and by ONR by contract N00014-86-K-0697.

Table I: Algorithmic performance Ref.[20].

PROBLEM	INPUT	SIMD MESH	SIMD HYPERCUBE
Matrix Multiply	$2N \times N$ matrices	$O(N)$	$O(\log N)$
Sorting	N^2 elements	$O(N)$	$O(\log^2 N)$
Connected Components	N vertex graph	$O(\sqrt{N})$	$O(\log^2 N)$
FFT	N element vector	$O(N)$	$O(\log N)$

Table II. Summary of Comparison.

Symbolic Substitution	POEM
Connection Topology/Grain Size <ul style="list-style-type: none"> • Mesh/very fine grain 	Connection Topology/Grain Size <ul style="list-style-type: none"> • Any connection topology including mesh/fine grain
Partial use of the advantages of optics: <ul style="list-style-type: none"> • Exploits parallelism and speed but not connectivity • Not expandable like VLSI mesh • Sensitive to technological defects 	Full use of major advantages of optics: <ul style="list-style-type: none"> • Exploits connectivity, parallelism and speed of optics • Limitation in signal skew in global irregular interconnections • Fault tolerant due to global interconnections
Addressable Memory: unknown <ul style="list-style-type: none"> • No RAM capability • Programmability: unknown difficulty • No possibility for space-time trade-offs 	Addressable Memory: RAM <ul style="list-style-type: none"> • MOS RAM, small storage cells, fast access • Programmability: conventional • Context switching capability
Hardware <ul style="list-style-type: none"> • Fast devices (ns-ps) • Larger power required (thresholding, splitting, shifting, combining and masking) • Large # of device and energy required to implement Boolean logic • May be OK for customized systems 	Hardware <ul style="list-style-type: none"> • Slower devices (100ns-100ps) • Small power dissipation per device • Smaller energy required per local interconnect and Boolean operations • Unsolved integration issues • OK for general purpose and special purpose.

References

1. J.W. Goodman, F.I. Leonberger, S.Y. Kung, and R.A. Athale, "Optical interconnections for VLSI systems," *Proc.IEEE*, 72, 850 (1984).
2. L.A. Bergman, W.H. Wu, A.R. Johnston, R. Nixon, S.C. Esener, C.C. Guest, P.K. Yu, T. Drabik, M.R. Feldman and S.H.Lee " Optical interconnects for VLSI" *Opt.Eng.* 25 (10), 1109 (1986)
3. W.H. Wu, L.A. Bergman, A.R. Johnston, C.C. Guest, S.C. Esener, P.K. Yu, M.R. Feldman and S.H.Lee " Implementation of optical interconnections for VLSI" *IEEE Trans. on Electron Devices*, 34 (3), 706 (1987).
4. R. Barakat and J. Reif, "Lower bounds on the computational efficiency of optical computing systems," *Appl.Opt.* 26, 1015 (1987)
5. F. Kiamilev, S. Esener, Y. Fainman and S.H. Lee, "Programmable opto-electronic multi-processors " To be published in *Appl. Opt.*
6. A. Huang, "Parallel Algorithms for Optical Digital Computers," in *IEEE Tenth International Optical Computing Conference*, 13 (1983)
7. K.-H. Brenner, A. Huang, and N. Streibl, "Digital Optical Computing with Symbolic Substitution," *Appl. Opt.* 25, 3054 (1986)
8. S.P. Kozaitis, "Higher-Ordered Rules for Symbolic Substitution," *Opt. Comm.* 65, 339 (1988)
9. S.H. Lee, S.C. Esener, M.A. Title, and T.J. Drabik, "Two-dimensional silicon/PLZT spatial light modulators: design considerations and technology," *Opt.Eng.* 25 (2), 250 (1986)
10. J.H. Wang, T.H.Lin, S.C. Esener, S. Dasgupta and S.H. Lee, "NMOS transistors fabricated by simultaneous laser-assisted crystallization and diffusion on silicon on electro-optic PLZT" *Proc. of M.R.S. symposium*, Boston MA,(1987).
11. M.R. Feldman, S.C. Esener, C.C. Guest, S.H. Lee, "Comparison between Optical and Electrical Interconnects based on Power and Speed Considerations," *Appl. Opt.* 27, 1742 (1988)
12. A.W. Burks, Ed., *Essays on Cellular Automata*, U. Illinois Press, Champaign-Urbana, (1970)
13. M.J. Murdocca, "Digital Optical Computing with one-rule cellular automata," *Appl. Opt.* 26, 682 (1987)
14. J.N. Mait, K.-H. Brenner, "Optical Symbolic Substitution: System Design using Phase-only Holograms," *Appl. Opt.* 27, 1692 (1988)
15. Karl-Heinz Brenner, "New implementation of symbolic substitution logic," *Appl. Opt.* 25(18), 3061 (1986)
16. W.C. Dally, *A Concurrent Data Structure for VLSI*, Ch.3, Kluwer (1987)
17. D.D.B. Miller, D.S. Chemla, T.C. Damen, T.H. Wood, C.A. Burrus, Jr., A.C. Gossard and W. Wiegmann, " The quantum well self-electro-optic effect device:optoelectronic bistability and oscillation and self-linearized modulation," *IEEE J. Quantum Electron.* QE-21, 1462 (1985)
18. C.D. Capps, R.A. Falk, T.L. Houk, "Optical Arithmetic/Logic Unit based on Residue Arithmetic and Symbolic Substitution," *Appl. Opt.* 27, 1682 (1988)
19. P. Ambs, Y. Fainman, S. Esener and S.H. Lee, "Holographic optical elements for SLM defect removal and for optical interconnects", submitted to *Appl.Opt.* (April 1988).

20. B. K. Jenkins, P. Chavel, R. Forcheimer, A.A. Sawchuk and T.C. Strand , "Architectural Implications of a digital optical processor", Appl.Opt. 23, 3465 (1984)
21. M.J. Quinn, Designing Efficient Algorithms for Parallel Computers (McGraw Hill, Inc. 1987)
22. R.M. Karp and V. Ramachandran, "A survey of parallel algorithms for shared-memory machines" UCB/CSD 88, 40 (1988)
23. T.J. Cloonan, "Performance Analysis of Optical Symbolic Substitution," Appl. Opt. 27, 1701 (1988)
24. S. D. Goodman and W. T. Rhodes, "Symbolic substitution applications to image processing", Appl. Opt. 27, 1708 (1988)
25. K.S. Huang, K. Jenkins and A.A. Sawchuk, " Binary image algebra representation of optical cellular logic and symbolic substitution", J. Opt.Soc.Am., 4 (12), 87 (1987)
26. J.Taboury, J.M.Wang, P. Chavel, F. Devos, P. Garda, "Optical Cellular Processor Architecture. 1. Principles," Appl. Opt. 27, 1643 (1988)

Figure Captions:

Fig. 1a. POEMs architecture.

Fig. 1b. Opto-electronic processing element array

Fig. 2. An example of a substitution rule using dual rail logic. Ref.[6]

Fig. 3. Murdocca's substitution rule. Ref.[13]

Fig. 4. Beamsplitter used for image replication and shifting

Fig. 5. Free space holographic interconnects

Fig. 6. Model for computing the energy dissipation in a symbolic substitution module.

Fig. 7. Model of an amplifying optical gate array.

Fig. 8. Model of a thresholding optical gate array.

Fig. 9. A flip-flop implemented with 50×56 pixels using Murdocca's rule.

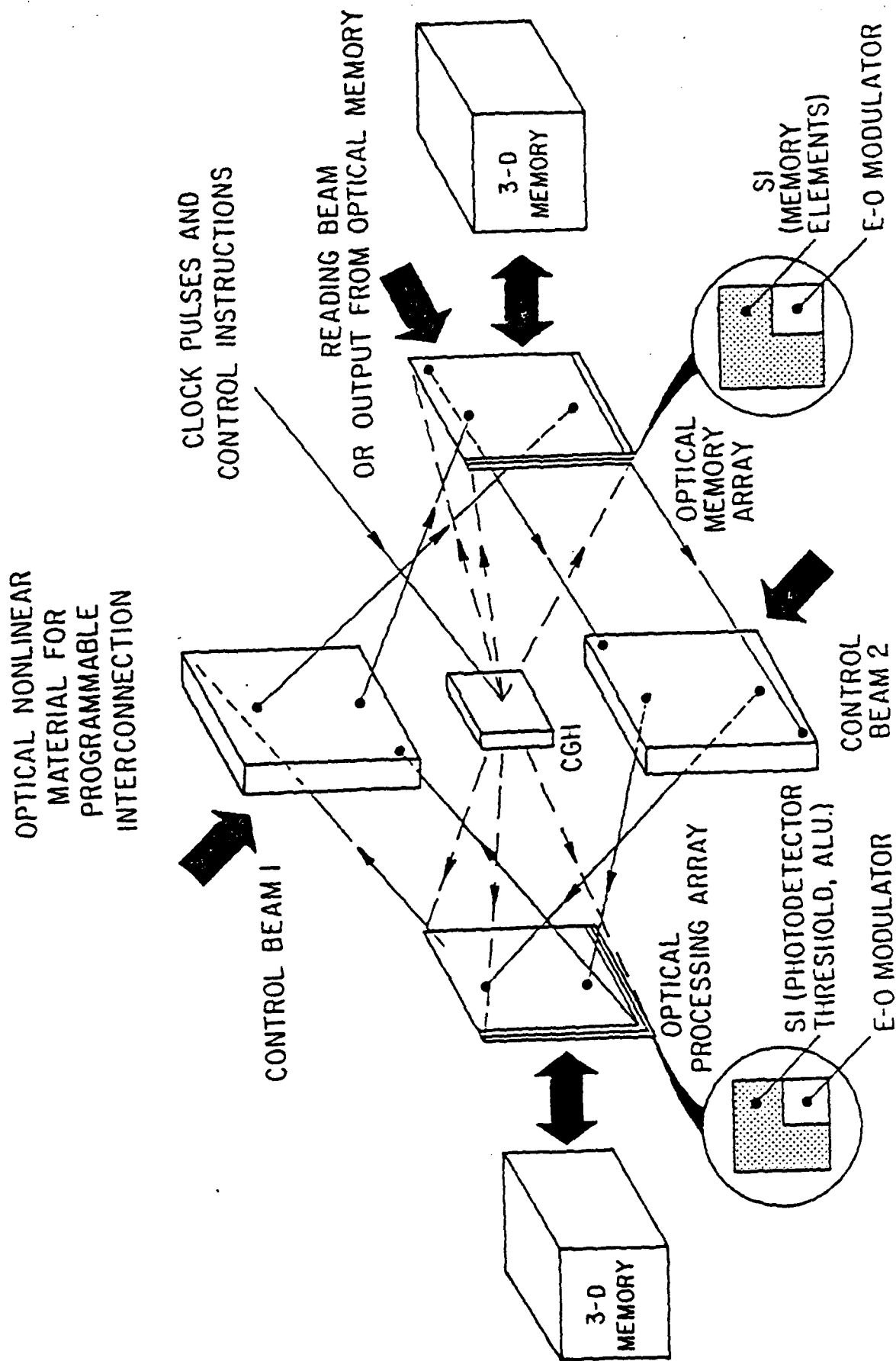


Fig. 1a. POEMs architecture.

Fig. 1a

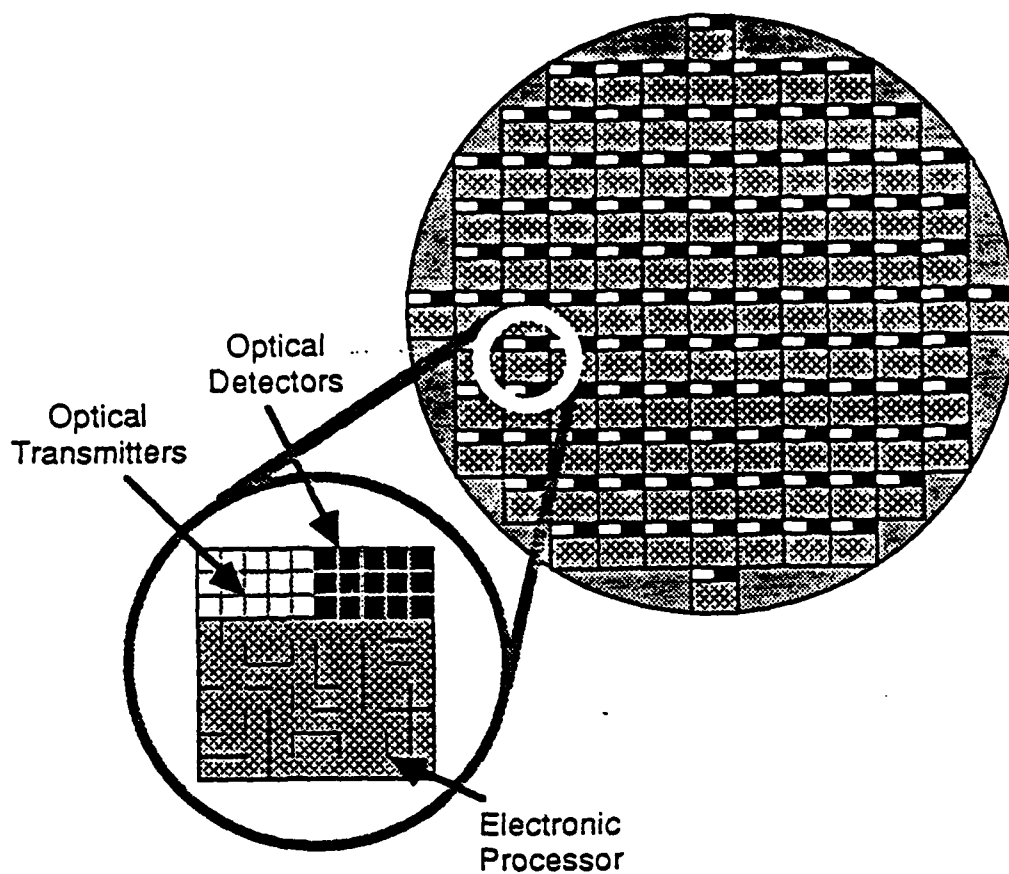


Fig. 1b. Opto-electronic processing element array

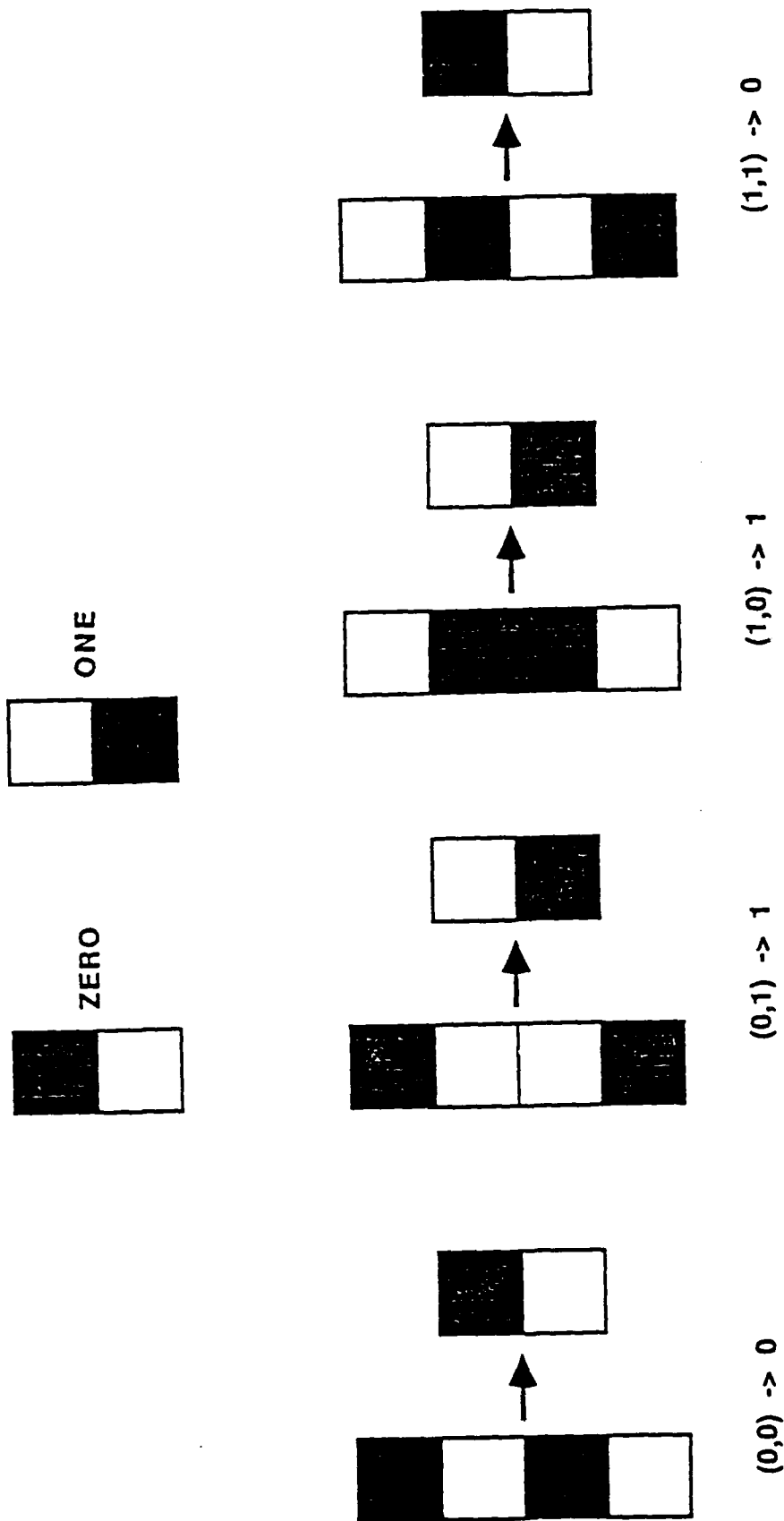
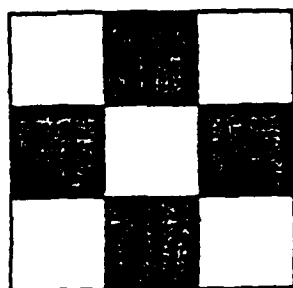


Fig. 2. An example of a substitution rule using dual rail logic. Ref.[6]

Left-Hand Side
(LHS)



Right-Hand Side
(RHS)

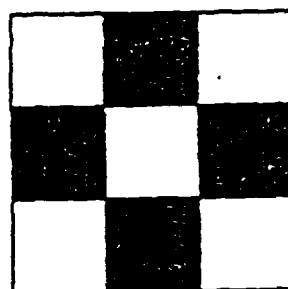
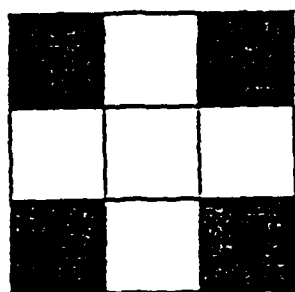
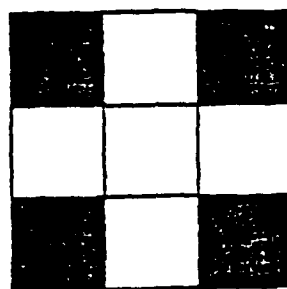


Fig. 3. Murdocca's substitution rule. Ref.[13]

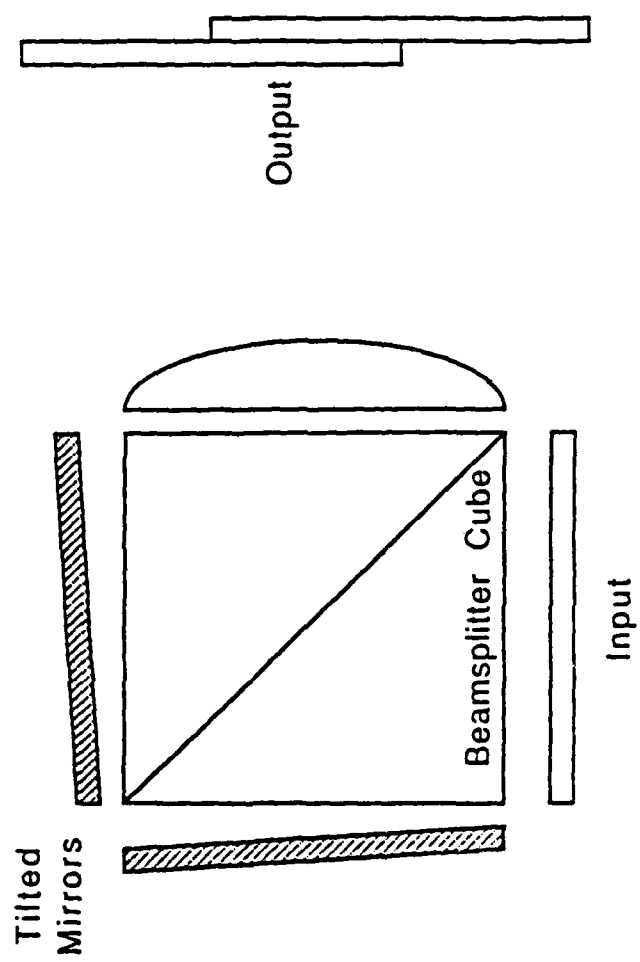


Fig. 4. Beamsplitter used for image replication and shifting

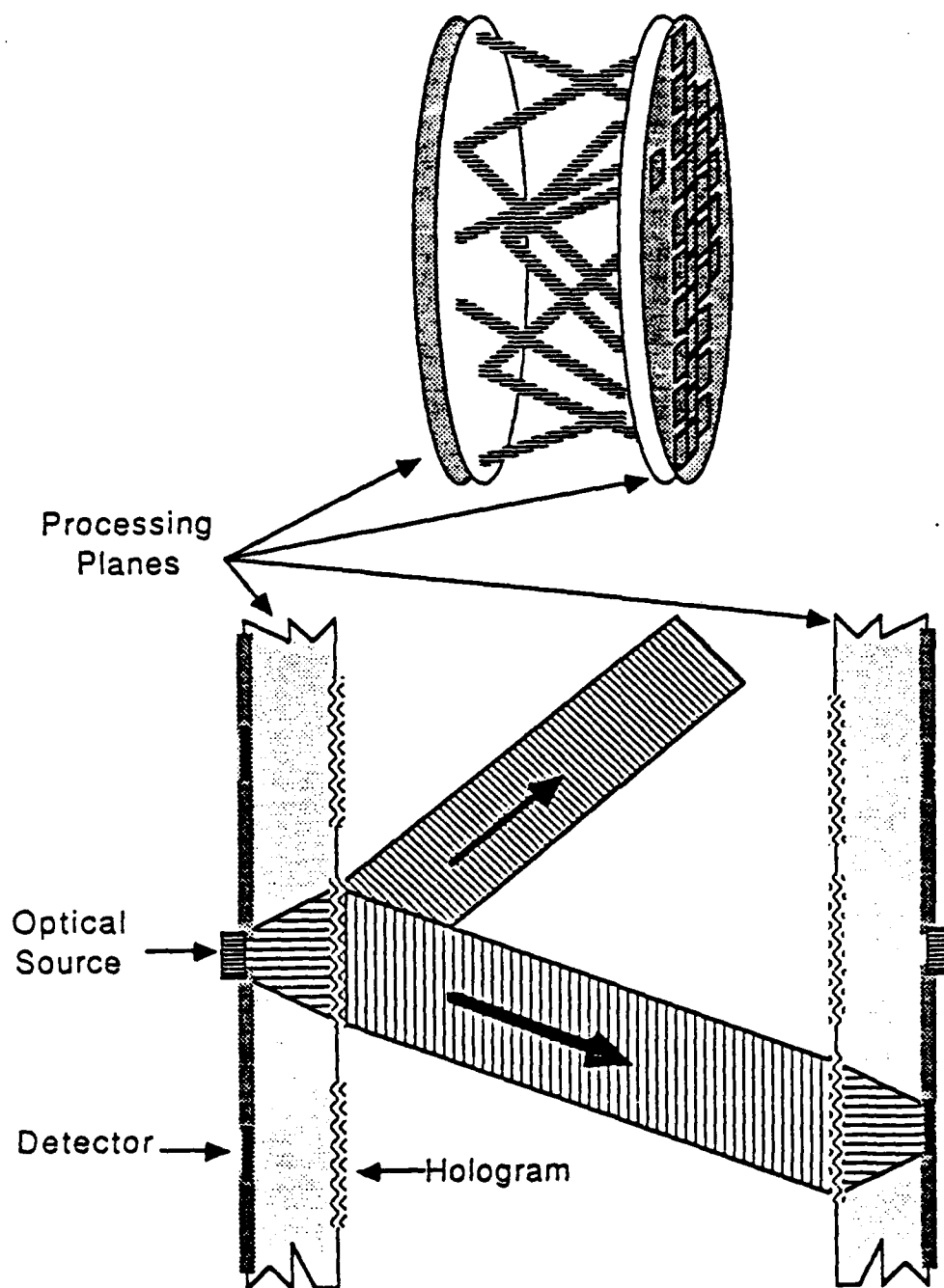


Fig. 5. Free space holographic interconnects

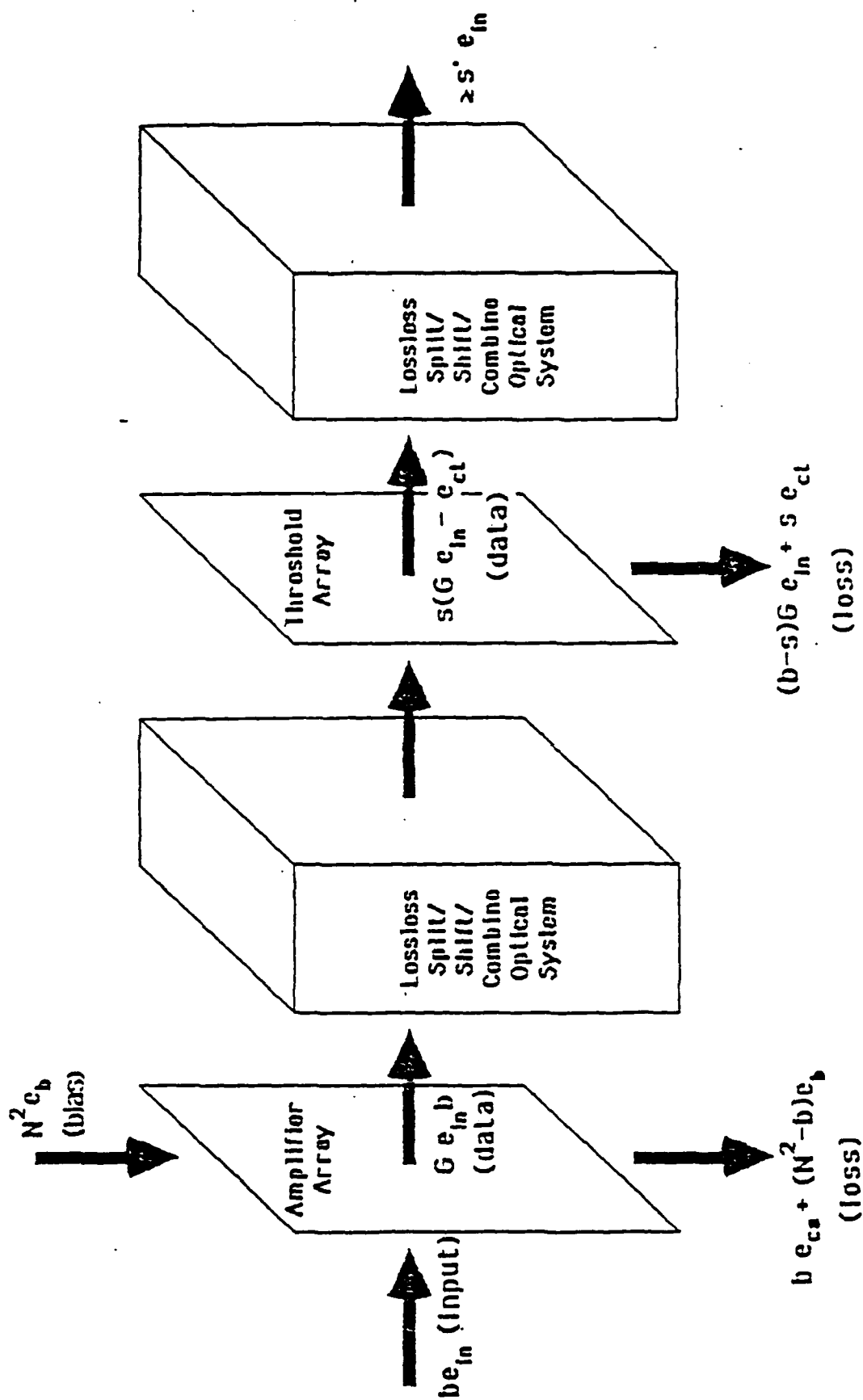


Fig. 6. Model for computing the energy dissipation in a symbolic substitution module.

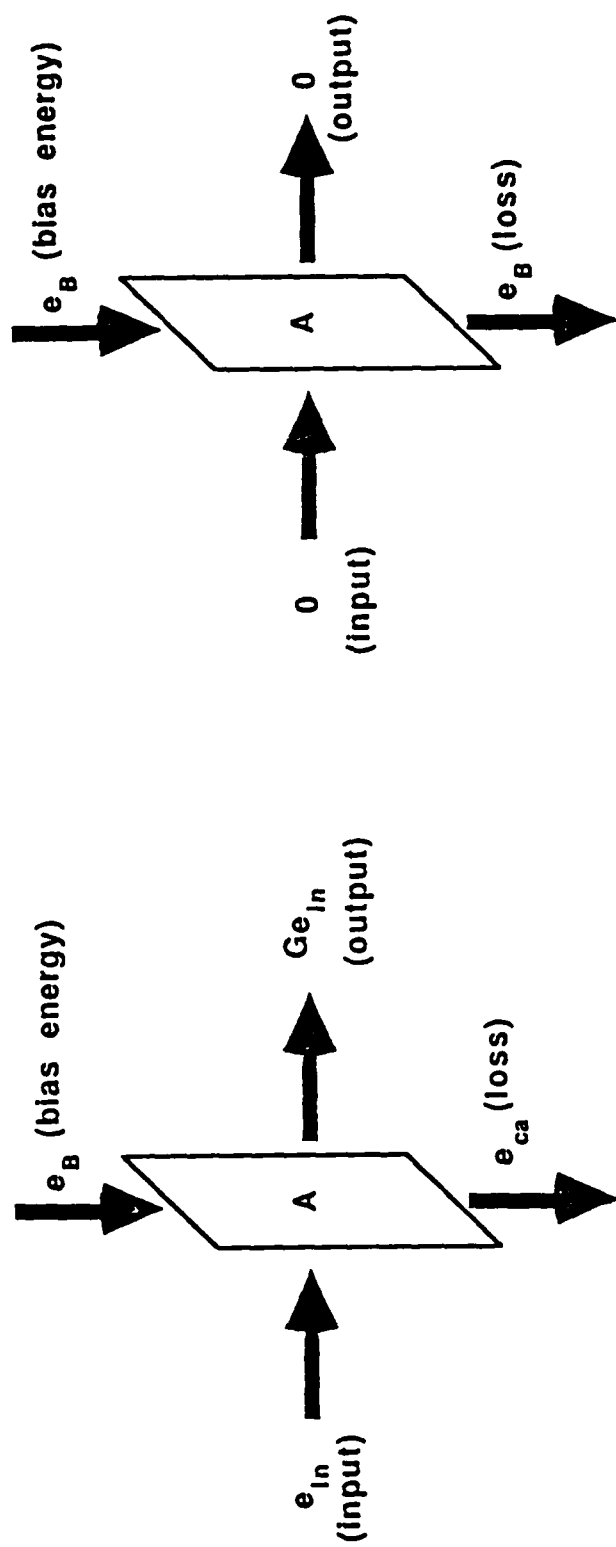


Fig. 7. Model of an amplifying optical gate array.

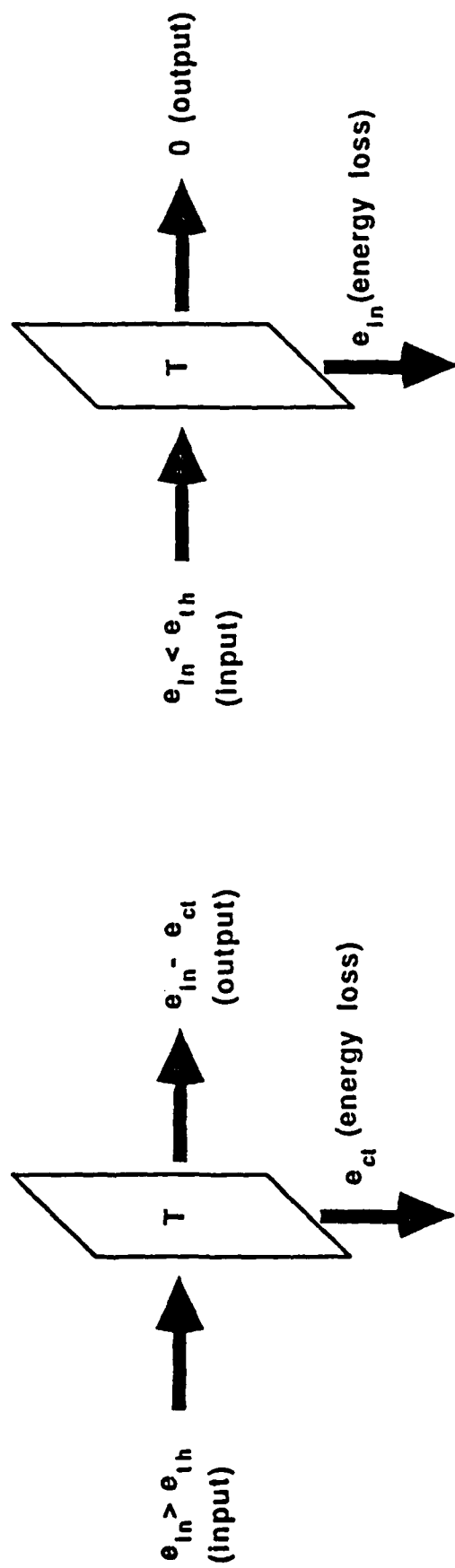


Fig. 8. Model of a thresholding optical gate array.

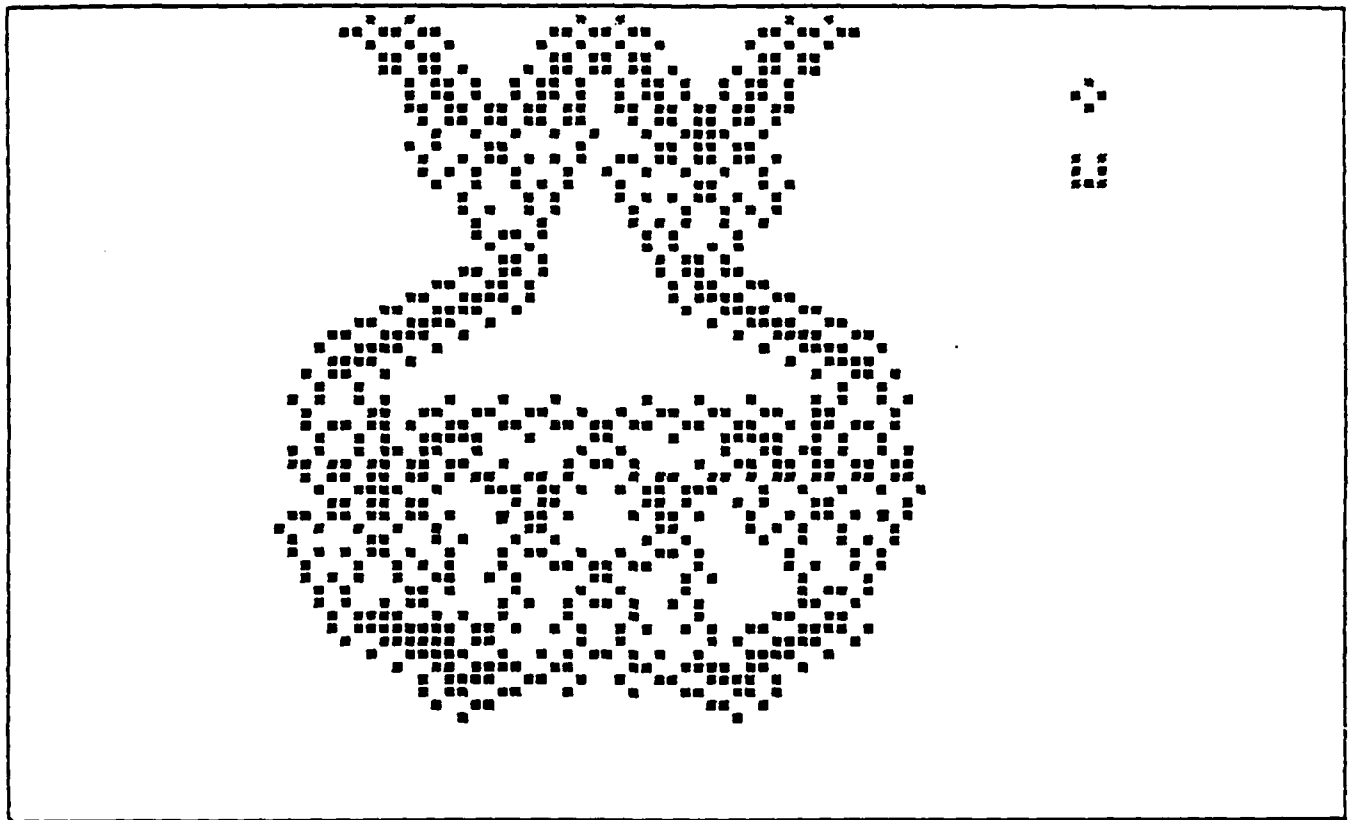


Fig. 9 A flip-flop implemented with 50 x 56 pixels using Murdocca's rule.